

CASCADE STRUCTURE FOR FINITE IMPULSE RESPONSE FILTERS AND LINEAR PREDICTION

Walter Humberto Orozco Tupacyupanqui^{1,*}

Abstract

This paper presents a complete analysis of the cascade structure for adaptive transversal filters based on adaptive algorithms. The standard structure of the cascade transversal FIR filter is obtained by replacing the whole structure by small ones with the same impulse response but having a less number of taps than the original structure. Computer simulation result shows the validity, reliability and the limitations that the model could have in its capacity of prediction. The optimal values of the model are compared with those obtained by the standard least mean square and recursive least square adaptive algorithms in order to verify the convergence of the weights and determine how fast this structure achieves those weights. For this case the speed of the algorithm is determined by the number of iterations that the filter requires to reach the minimum square value of its learning curve.

Keywords: Cascade structures, Finite impulse response filters, Transversal filters, Linear estimation, LMS algorithm, RLS algorithm.

Resumen

Este artículo presenta un análisis completo de la estructura en cascada para filtros transversales basados en algoritmos adaptativos. La estructura general de un filtro transversal FIR en cascada se obtiene reemplazando toda la estructura por pequeñas secciones independientes de filtros de la misma respuesta al impulso pero con un número inferior de pesos al original. Los resultados de la simulación en computadora muestran la validez, capacidad y las limitaciones que el modelo podría tener en su capacidad de predicción. Los valores óptimos del modelo son comparados con aquellos obtenidos por los algoritmos adaptativos convencionales de mínimo cuadrado medio y mínimo cuadrado medio recursivo para verificar la convergencia de los pesos y determinar qué tan rápido la estructura alcanza dichos pesos. En este caso la rapidez del algoritmo se determina en base al número de iteraciones que el filtro requiere para llegar al valor mínimo cuadrado de su curva de aprendizaje.

Palabras clave: estructura en cascada, filtros de respuesta al impulso finita, filtros transversales, estimación lineal, algoritmo LMS, algoritmo RLS.

^{1,*} *Máster en Ciencias en Ingeniería Eléctrica y Sistemas, Ingeniero Eléctrico, Estudiante de Doctorado en Comunicaciones y Electrónica de la Sección de Estudios de Posgrado e Investigación del Instituto Politécnico Nacional de México. Docente de la carrera de Ingeniería Eléctrica y Electrónica de la Universidad Politécnica Salesiana, sede Cuenca. Autor para correspondencia ✉: worozco@ups.edu.ec*

Recibido: 30 - Octubre - 2012, Aprobado tras revisión: 18 - Noviembre - 2012

Forma sugerida de citación: Orozco, W. (2012). "Cascade Structure for Finite Impulse Response Filters and Linear Prediction". *INGENIUS*. N.º8, (Julio/Diciembre). pp. 20-29. ISSN: 1390-650X

1. Introduction

Adaptive systems are commonly used to exploit the spatial domain in an attempt to enhance coverage and capacity by minimizing interference in different types of engineering systems. For example, in an adaptive beamformer, array shading coefficients are adjusted to maximize the signal to noise ratio so that the beam pattern has its nulls in the direction of major noise sources.

In adaptive predictors and echo cancellers, the coefficients are adjusted to minimize prediction error and undesired interferences in the mean square sense. The essential and principal property of these systems is its time-varying, self-adjusting performance [1].

Over the last sixty years, adaptive systems have been actively researched. One of the pioneers in developing adaptive theory was Widrow who used the concept of adaptive linear combiner together with a set of adjusted weights to reduce the mean-square error [1],[2]. Different types of adaptive algorithms, including the Least-Mean-Square (LMS) and Recursive-Least-Square (RLS), have been developed and applied to areas such as smart antennas, speech communications, electrocardiography, and seismic signal processing. Adaptive noise canceling is in fact applicable to a wide variety of signal-enhancement situations, because noise characteristics are not often stationary in real-world situations [3].

A lot of research about new adaptive structure filters and algorithms are carried out to enhance their performance. One of these structures is the cascade form [4],[5], [6], many applications and analyses have been developed and accomplished to give a better understanding of this implementation, for example in [7] an infinite impulse response (IIR) adaptive filter algorithm is constructed as a cascade of transversal structure as well as all pole lattice structure.

The filter structure is adapted using the simple hyperstable adaptive recursive filter (SHARF) algorithm in which the transversal and lattice sections are updated using the simultaneous perturbation method. This algorithm preserves the stability of the IIR adaptive filter during the adaptation with lower computational complexity and cost than conventional algorithms. In [8] a general-purpose algorithm was proposed for the design of linear phase finite impulse response (FIR) filters in cascade form with discrete coefficients.

This algorithm decomposes the overall filter into subfilters during the traverse of a tree search of the overall filter. The optimization complexity is of the same order as the single-stage filter optimization. In [9] the use of different high-level filter architectures

(cascade, lattice, and parallel) is analyzed and their advantages and disadvantages are discussed. This information serves as a useful comparative analysis in the selection of high-level filter architecture for a particular problem. In this paper we analyze the optimal values obtained by changing the structure of a transversal filter by one which is based on cascade structures for linear prediction problems.

In Section 2, a complete development of the equations has been made in order to obtain a sufficient understanding of the structure and its limitations. Section 3 describes the equations related to the simplest cascade structure and how its constrained equations have relationship with the standard transversal filter. Section 4 describes the adaptive algorithms that will be used in this paper for the cascade structure to contrast the results obtained in the previous section. Results obtained from computer simulation for two stages and one tap for stage are presented in Section 5. Finally, Section 6 concludes the paper.

2. Linear prediction and the cascade structure

The problem of linear prediction is analogous to that of finding the whitening filter for a given signal. FIR adaptive filters are suitable structures for this particular problem, especially when dealing with only locally stationary signals. The adaptive filter tries to generate an estimate of the current input sample for a linear combination of a determined number of past samples. Unfortunately, large values of the predictor's length increase the eigenvalue spread effects and force the use of a smaller step size, which implies a slower convergence. A possible way to counteract this problem is to replace the whitening filter by a series of sequential refining of the prediction performed by a cascade of small-order whitening filters [6].

Figure 1 shows the structure of the cascade for one-step-ahead linear prediction problem. The global adaptive filter is replaced by one of various sections and each section is independent among them [1],[5].

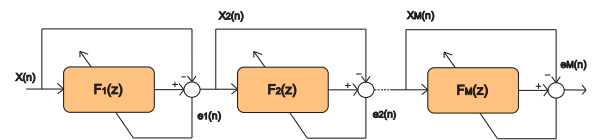


Figure 1. Standard structure of the cascade predictor.

The analysis of the cascade structure is developed by considering the simplest form of the cascade predictor which is composed of one weight for stage as

shown in Figure 2 [6],[10]. Regardless of the particular algorithm governing the adaptation, the following equation rules its behavior for each stage

$$\hat{x}_k(n) = \sum_{m=1}^M w_k^{(m)} x(n-m) \quad (1)$$

$$e_k(n) = x_k(n) - \hat{x}_k(n) \quad (2)$$

By substituting (1) into (2) yields

$$e_k(n) = x_k(n) - \sum_{m=1}^M w_k^{(m)} x(n-m) \quad (3)$$

For this particular analysis $M = 1$ for stage, thus

$$e_k(n) = x_k(n) - w_k^{(1)} x(n-1) \quad (4)$$

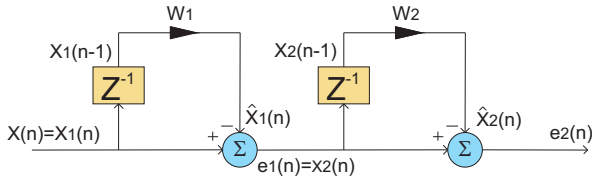


Figure 2. The simplest cascade predictor for two stages and only one weight for stage.

Equation (4) represents the output of each stage and at the same time the approximation error, where $x_{k+1}(n) = e_k(n)$ are the adjustable weights of the k -th predictor. It is possible to omit superscript in (4) for this particular case due to there is only one tap for stage. The stages are connected in cascade so that

$$x_{k+1}(n) = e_k(n) \quad (5)$$

In the adaptation process with performance feedback, the weight vector is adjusted to cause the output to agree as closely as possible with the desired signal. This is accomplished by comparing the output with the desired response to obtain an error signal and then adjusting the tap vector to minimize this error. In most practical instances the adaptive process is oriented toward minimizing the mean-square value [1],[3],[11] taking the square and expected value of (4), we have

$$e_k^2(n) = x_k^2(n) - 2w_k x_k(n) x(n-1) + \dots \\ \dots + w_k^2 x_k^2(n-1) \quad (6)$$

$$E[e_k^2(n)] = E[x_k^2(n)] - \dots$$

$$\dots - 2w_k E[x_k(n) x_k(n-1)] + w_k^2 E[x_k^2(n-1)] \quad (7)$$

Equation (7) is known as the mean-square error of the system. Many useful adaptive processes rely on the weight vector to seek the minimum of the performance surface by gradient methods. The gradient of the mean-square-error performance surface is denoted as ∇ , which can be obtained by differentiating the last equation (7) with respect to the coefficients of each section [2],[3],[10] thus

$$\nabla = \frac{\partial E[x_k^2(n)]}{\partial w_k} \quad (8)$$

$$\nabla = -2E[x_k(n) x_k(n-1)] + 2w_k E[x_k^2(n-1)] \quad (9)$$

If it is assumed convergence towards the Wiener-Hopf solution for the single stage, $w_k \rightarrow w_k^*$; thus

$$0 = -2E[x_k(n) x_k(n-1)] + 2w_k^* E[x_k^2(n-1)] \quad (10)$$

$$w_k^* = \frac{E[x_k(n) x_k(n-1)]}{E[x_k^2(n-1)]} \quad (11)$$

The autocorrelation function [11] between two signals is given by

$$E[x_k(n-p) x_k(n-i)] = r(i-p) \quad (12)$$

Substituting (12) in (11), the optimal value of the weight for the simplest model can be written as a function of the autocorrelation, therefore

$$w_k^* = \frac{r_k(1)}{r_k(0)} \quad (13)$$

The next step is to consider (5) in order to develop the equations for the $k+1$ stage and determine its relationship with the previous stage, therefore the autocorrelation of the output at the k -th stage can be expressed as

$$E[e_k(n) e_k(n-m)] = E[x_{k+1}(n) x_{k+1}(n-m)] \quad (14)$$

Assuming convergence of the weight to the optimal value, the left side of (14) after some straightforward manipulation can be written as

$$r_k(m) - w_k^* r_k(m+1) - w_k^* r_k(m-1) + \dots \\ \dots + (w_k^*)^2 r_k(m) \quad (15)$$

Developing a similar manipulation to obtain (15), the right side of (14) can be written as a function of the correlation as

$$E[x_{k+1}(n) x_{k+1}(n-m)] = r_{k+1}(m) \quad (16)$$

Therefore, (16) can be expressed as

$$\left[1 + (w_k^*)^2\right] r_k(m) - w_k^* [r_k(m+1) + r_k(m-1)] \quad (17)$$

In particular for $m = 0$, from (17) gives

$$r_{k+1}(0) = \left[1 + (w_k^*)^2\right] r_k(0) - w_k^* [r_k(1) + r_k(-1)] \quad (18)$$

But

$$E[x_k(n) x_k(n-i)] = E[x_k(n-i) x_k(n)] \quad (19)$$

$$r(i) = r(-i) \quad (20)$$

By substituting (20) in (18) yields

$$r_{k+1}(0) = \left[1 + (w_k^*)^2\right] r_k(0) - 2w_k^* r_k(1) \quad (21)$$

Equation (21) may be expressed by substituting (13) in it, therefore

$$r_{k+1}(0) = r_k(0) + \frac{r_k^2(1)}{r_k(0)} - 2\frac{r_k^2(0)}{r_k(0)} \quad (22)$$

$$r_{k+1}(0) = r_k(0) - \frac{r_k^2(1)}{r_k(0)} \quad (23)$$

The performance of the system depends on whether or not the mean-square-error has the possibility to increase as a function of the number of stages; therefore the MSE is defined as

$$\psi_k = E[e_k^2(n)] \quad (24)$$

$$\psi_k = E[e_k(n) e_k(n)] \quad (25)$$

$$\psi_k = E[x_{k+1}(n) e_{k+1}(n)] \quad (26)$$

$$\psi_k = r_{k+1}(0) \quad (27)$$

Substituting (23) in (27) yields

$$\psi_k = r_k(0) - \frac{r_k^2(1)}{r_k(0)} \quad (28)$$

If it is supposed $\psi_k = r_{k+1}(0)$ then

$$\psi_{k-1} = r_k(0) \quad (29)$$

Substituting (29) into (28) results in

$$\psi_k = \psi_{k-1} - \frac{r_k^2(1)}{\psi_{k-1}} \quad (30)$$

Therefore,

$$\psi_k \leq \psi_{k-1} \quad (31)$$

According to (31) the final MSE is not an increasing function of the number of stages.

3. The simplest two stages cascade structure

For a two-stage cascade filter exact expressions for the weights can be derived in an explicit manner. Even though, this is a very simple application of the Wiener solution, it can be manageable. For the linear filter with two taps, (3) is expanded for $k = 1, 2$; therefore

$$e_1(n) = x_1(n) - w_1 x(n-1) \quad (32)$$

$$e_2(n) = x_2(n) - w_2 x(n-2) \quad (33)$$

By (5), the following equation is obtained

$$x_2(n) = e_1(n) \quad (34)$$

$$x_2(n) = x_1(n) - w_1 x(n-1) \quad (35)$$

Substituting (34) into (32) yields

$$e_2(n) = x_1(n) - w_1 x(n-1) + w_2 x(n-2) \quad (36)$$

Taking the Z-transform of (36), and considering that $x_1(n) = x(n)$ after some straight manipulation

$$F(z) = \frac{E_2(z)}{X(z)} = 1 - w_1 z^{-1} + w_2 z^{-2} \quad (37)$$

Considering convergence to the Wiener solution [2] of this filter, (37) can be expressed as

$$F^*(z) = 1 - w_1^* z^{-1} + w_2^* z^{-2} \quad (38)$$

From the Wiener solution [1] for a standard transversal filter having two weights, the gradient can be calculated as

$$\nabla = 2 \begin{bmatrix} r(0) & r(1) \\ r(1) & r(0) \end{bmatrix} \begin{bmatrix} w_{n1} \\ w_{n2} \end{bmatrix} - 2 \begin{bmatrix} r(1) \\ r(2) \end{bmatrix} \quad (39)$$

In order to obtain the optimal solution, the gradients must be zero, after some straightforward manipulations the optimal weights are expressed as

$$w_{n1}^* = \frac{r(1)[r(0) - r(2)]}{r(0)^2 - r(1)^2} \quad (40)$$

$$w_{n2}^* = \frac{r(0)r(2) - r(1)^2}{r(0)^2 - r(1)^2} \quad (41)$$

According to [7], the cascade weights for this system can be defined as

$$w_1^* = w_{c1}^* + w_{c2}^* \quad (42)$$

$$w_2^* = -w_{c1}^* w_{c2}^* \quad (43)$$

Therefore, (38) can be written as

$$F^*(z) = 1 - (w_{c1}^* + w_{c2}^*)z^{-1} - w_{c1}^* w_{c2}^* z^{-2} \quad (44)$$

From (13), for each stage the optimal weights can be written as a function of the autocorrelation function as

$$w_{c1}^* = \frac{r_1(1)}{r_2(0)} \quad (45)$$

$$w_{c1}^* = \frac{r_2(1)}{r_2(0)} \quad (46)$$

Substituting (45) into (17) with $k = 1$ and $m = 1$, after some straightforward manipulation

$$r_2(1) = r_1(1) \left[\frac{r_1^2(1)}{r_1^2(0)} - \frac{r_1(0)r_1(2)}{r_1^2(0)} \right] \quad (47)$$

From equation (23), (46) and (47) the optimal values for the cascade structure can be expressed as

$$w_{c2}^* = \frac{r_1(1)}{r_2(0)} \left[\frac{r_1^2(1)}{r_1^2(0)} - \frac{r_1(0)r_1(2)}{r_1^2(0)} \right] \quad (48)$$

$$w_{c2}^* = \frac{r_1(1)}{r_1(0)} \left[\frac{r_1^2(1) - r_1(0)r_1(2)}{r_1^2(0)} \right] \quad (49)$$

Finally, substituting (45), (49) into (42) and (43) yields

$$w_1^* = \frac{r(1)[r(0) - r(2)]}{r(0)^2 - r(1)^2} = w_{n1}^* \quad (50)$$

$$w_2^* = \frac{r(1)^2}{r(0)^2} \left[\frac{r(1)^2 - r(0)r(2)}{r(0)^2 - r(1)^2} \right] = \delta^2 w_{n2}^* \quad (51)$$

According to (50) and (51) the cascade weights converge to a biased version of the optimal filter [5]; in this case the bias affects the second coefficient as a function of the autocorrelation of the signals.

4. Adaptive algorithms for learning

In addition to (50) and (51), some adaptive algorithms have been used to obtain the optimal values for the cascade form. In this section we describe in simple form its algorithms.

4.1 LMS algorithm

The optimal weight vector \mathbf{W} that solves the minimum-square error can be updated according to [3],[11].

$$\min_w E[\mathbf{d} - \mathbf{u}\mathbf{w}] \quad (52)$$

$$\omega_i = \omega_{i-1} + \mu x_i^* [d(i) - x_i \omega_{i-1}] \quad (53)$$

where μ is a positive step-size and \mathbf{d} is the desired signal.

4.2 RLS algorithm

The optimal weight vector \mathbf{W} that solves (52) can be updated according to [3],[11].

$$P_i = \lambda^{-1} \left[P_{i-1} - \frac{\lambda^{-1} P_{i-1} u_i^* u_i P_{i-1}}{1 + \lambda^{-1} u_i P_{i-1} u_i^*} \right] \quad (54)$$

$$\omega_i = \omega_{i-1} + P_i u_i^* [d(i) - u_i \omega_{i-1}] \quad (55)$$

where $0 < \lambda \leq 1$ and $P_{-1} = \epsilon^{-1} \mathbf{I}$ for some small positive scalar ϵ and \mathbf{d} is the desired signal.

5. Computer simulations

The ability of the cascade linear predictor to estimate a signal that has been corrupted by noise is investigated by means of MATLAB[®] simulations. The main purpose of the simulation is to analyze the behavior of the biased version of the second weight of the cascade form and how fast this architecture is. For comparison purposes, results have been obtained for LMS and RLS algorithms. For the simulation, the following parameters have been used.

- The random samples of a sinusoidal signal are independent.
- A random variable r has been defined as $\mathbf{r} = k\mathbf{v}$ where k is a constant and \mathbf{v} is a normal random variable with mean zero and variance 1.
- The samples for cycle (N) have been increased from 8 to 20 by increments of 1.

- Three different variances of \mathbf{r} have been taken: 0.01, 0.04 and 0.09.

For this problem the signal that has been degraded by noise is defined as

$$x(n) = \sin\left(\frac{2\pi n}{N}\right) + r_n \quad (56)$$

According to [1],[6], the correlation matrix and the cross-correlation vector are defined respectively as

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) \\ r(1) & r(0) \end{bmatrix} \quad (57)$$

$$\mathbf{P} = \begin{bmatrix} r(1) \\ r(2) \end{bmatrix} \quad (58)$$

Where $r(0)$, $r(1)$ and $r(2)$ are defined as

$$r(0) = 0,5 + \phi \quad (59)$$

$$r(n) = 0,5\cos\left(\frac{2\pi n}{N}\right) \quad (60)$$

After some straight mathematical procedure, the correlation and cross-correlation (for a variance of \mathbf{r} equals 0.01 and $N = 20$), respectively are

$$\mathbf{R} = \begin{bmatrix} 0,51 & 0,4755 \\ 0,4755 & 0,51 \end{bmatrix} \quad (61)$$

$$\mathbf{P} = \begin{bmatrix} 0,4755 \\ 0,4045 \end{bmatrix} \quad (62)$$

The optimal solution given by Widrow [1], is defined as

$$\mathbf{W}^* = \mathbf{R}^{-1} \mathbf{P} \quad (63)$$

$$\mathbf{W}^* = \begin{bmatrix} 1,4755 \\ -0,5825 \end{bmatrix} \quad (64)$$

By (50) and (51) the optimal solution for this example must be

$$\mathbf{W}_c^* = \begin{bmatrix} 1,4755 \\ -0,5064 \end{bmatrix} \quad (65)$$

Figure 3 shows the convergence of the weights for the cascade system and the adaptive LMS algorithm used in this paper. For the simulation 8 samples for cycle have been taken in order to get a bias between the Wiener solution and the cascade solution. Even though the program has been ran for 1250 iterations only the first 500 iterations have been shown for this Figure.

Figure 4 shows the convergence of the weights for the cascade system and the adaptive LMS algorithm,

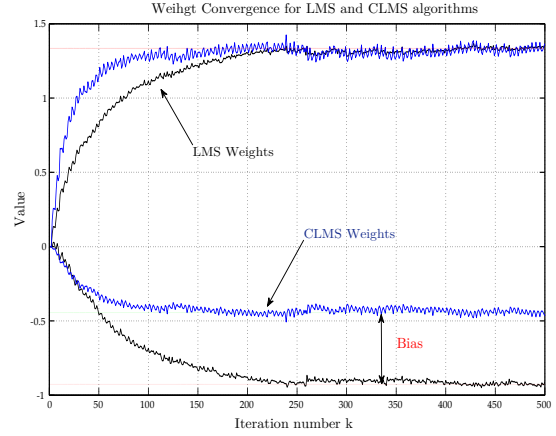


Figure 3. Convergence of the weights for the LMS, CLMS and Wiener solution for 8 samples for cycle and noise variance of 0.01.

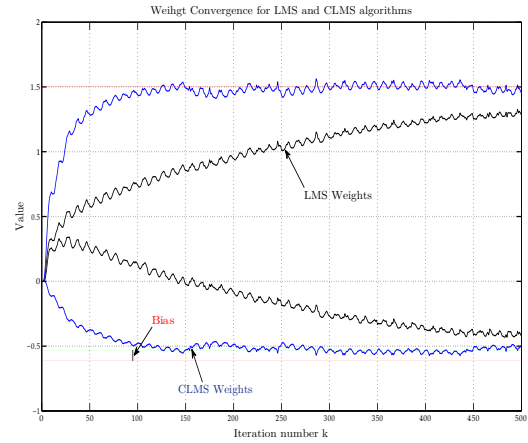


Figure 4. Convergence of the weights for the LMS, CLMS and Wiener solution for 20 samples for cycle and noise variance of 0.01.

but 20 samples for cycle have been taken to get a different bias between the Wiener solution and the cascade solution.

Figure 5 shows the convergence of the weights for the cascade system and the adaptive RLS algorithm. For the simulation 8 samples for cycle have been taken. Even though the program has been ran for 1250 iterations only the first 600 iterations have been shown for this Figure.

Figure 6 shows the convergence of the weights for the cascade system and the adaptive RLS algorithm, but 20 samples for cycle have been taken to get a different bias between the Wiener solution and the cascade solution.

Figure 7 shows the behavior of the bias when the samples for cycle have been increased. As a consequence of this change, the bias shows an exponential decay.

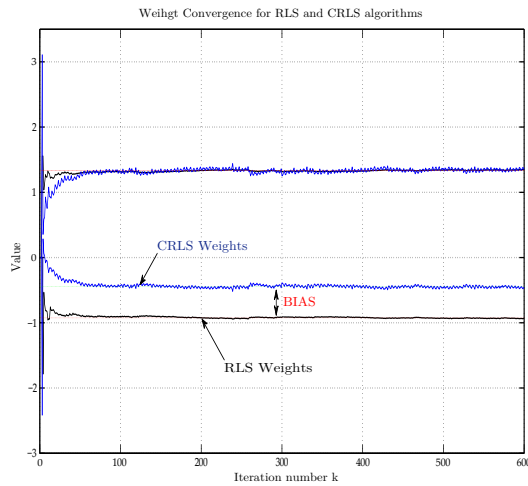


Figure 5. Convergence of the weights for the RLS, CRLS and Wiener solution for 8 samples for cycle and noise variance of 0.01.

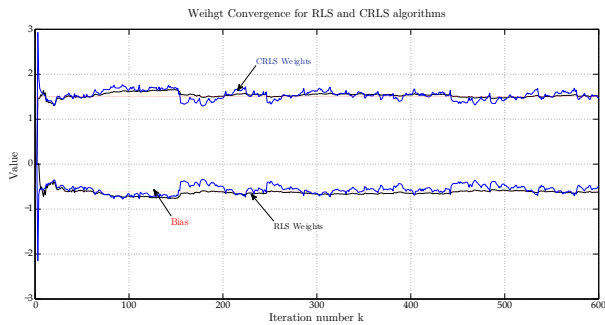


Figure 6. Convergence of the weights for the RLS, CRLS and Wiener solution for 20 samples for cycle and noise variance of 0.01.

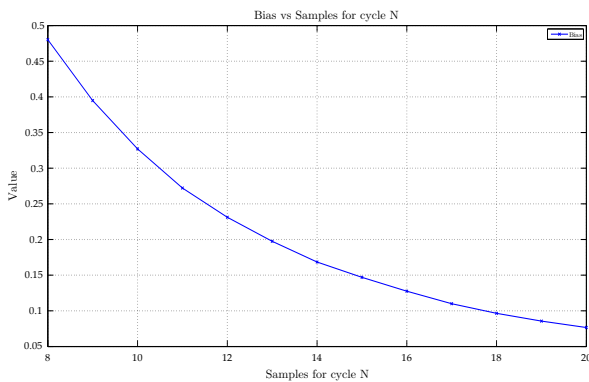


Figure 7. Bias versus the samples for cycle N for CLMS and CRLS algorithms for a noise variance of r equals to 0.01.

Figure 8 shows the learning curve of each adaptive algorithm for 8 samples for cycle. For each N, the algorithms reach the minimum value which is near

zero. Only the first 450 iterations have been shown for this Figure.

Figures 9 shows the number of iterations required to reach the optimal weights to minimize the mean-square error. According to the results, the cascade LMS is faster than the LMS; however, the cascade RLS is a little bit slower than the regular RLS.

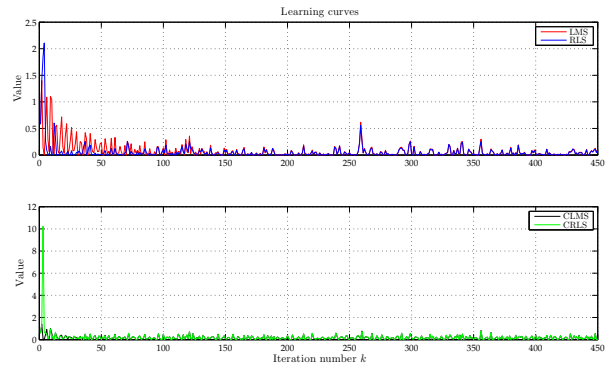


Figure 8. The learning curves of the LMS, RLS, CLMS and CRLS adaptive algorithms for 8 samples for cycle and noise variance of 0.01.

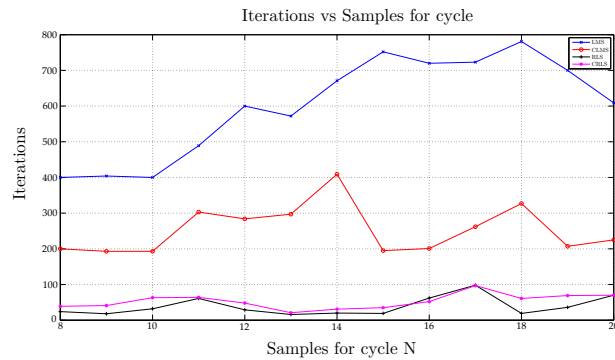


Figure 9. Iterations versus Samples for cycle for the LMS, RLS, CLMS and CRLS adaptive algorithms for 8 samples for cycle and noise variance of 0.01.

Figure 10 shows the convergence of the weights for the cascade system and the adaptive LMS algorithm for a noise variance of 0.04; meanwhile, Figure 11 shows the convergence of the weights for a noise variance of 0.09.

Figure 12 shows the convergence of the weights for the cascade system and the adaptive RLS algorithm for a noise variance of 0.04; meanwhile, Figure 13 shows the convergence of the weights for a noise variance of 0.09.

Figure 14 shows the changes of the bias of the second weights for the cascade system when the noise variance has been increased as well as the samples for

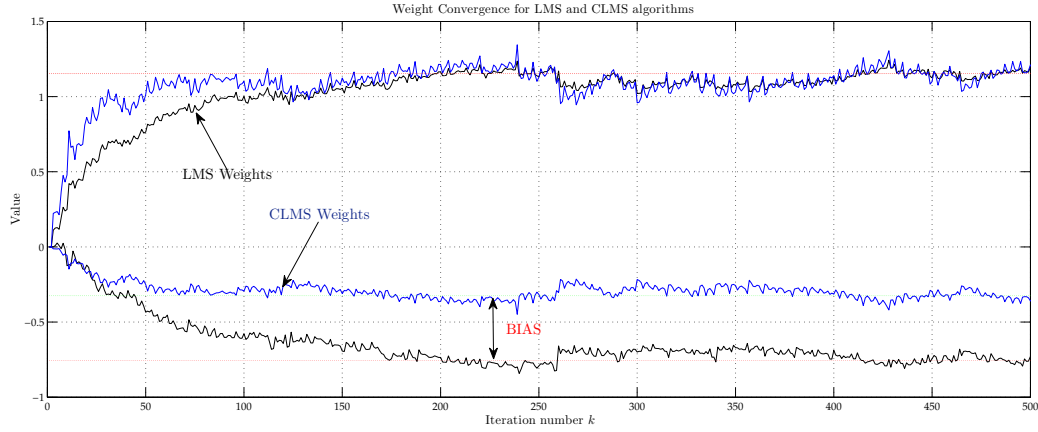


Figure 10. Convergence of the weights for the LMS, CLMS and Wiener solution for 8 samples for cycle and noise variance of 0.04.

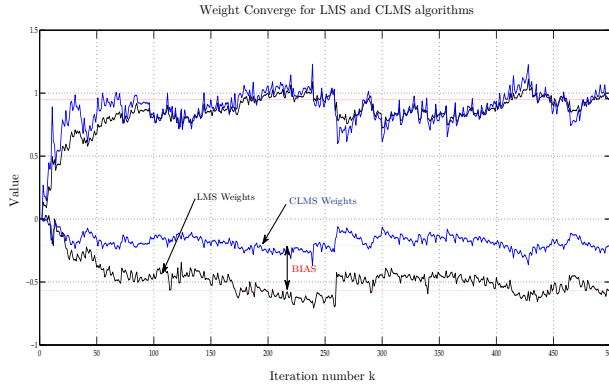


Figure 11. Convergence of the weights for the LMS, CLMS and Wiener solution for 8 samples for cycle and noise variance of 0.09.

cycle. As a consequence of these changes, the bias is smaller.

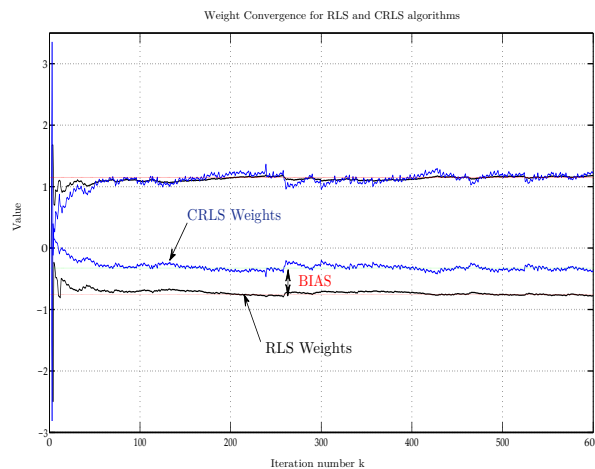


Figure 12. Convergence of the weights for the RLS, CRLS and Wiener solution for 8 samples for cycle and noise variance of 0.04.

6. Conclusions

The Cascade-Least-Mean-Square (CLMS) algorithm displays the ability to converge to the optimal solution (Wiener) much faster than the LMS algorithm for the same equivalent filter. This condition remains, even though the number of samples for cycle (N) and the variance noise have been changed. On the other hand, the Cascade-Recursive-Least-Square (CRLS) algorithm is a little bit slower than the RLS algorithm for the same changes. That is to say, the cascade form of the recursive algorithm does not enhance the performance of the standard RLS filter.

In general, for both algorithms the second weight converges to a biased version of the direct form filter. This bias is smaller when the signal has a narrow spectrum of samples of the frequency. This bias can be reduced as much as it is possible, but it cannot be completely eliminated. Another quality of the bias is referred to its dependence on the variance of the noise. When the variance of the noise has been increased the bias is smaller.

The number of iterations required to reach the optimal weight is almost the same, even though the noise variance has been increased. Nevertheless, the noise effects deteriorate the behavior of the weights towards the permanent optimum values. This characteristic is easily noticed for LMS algorithms.

For this type of filters, the weights for each stage does not have the ability to converge to the optimal value by itself; the reason is because each stage only has one tap; however, the final result of the equivalent cascade form depends on the weight in each stage. The first weight in the equivalent cascade form is the addition of the weights for separated; meanwhile, the second optimal weight is the negative multiplication.

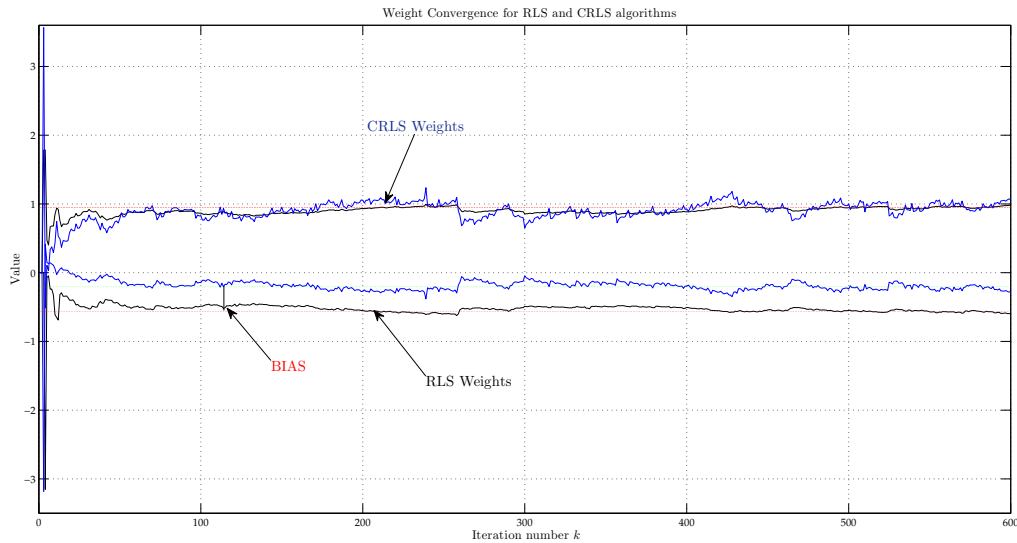


Figure 13. Convergence of the weights for the RLS, CRLS and Wiener solution for 8 samples for cycle and noise variance of 0.04.

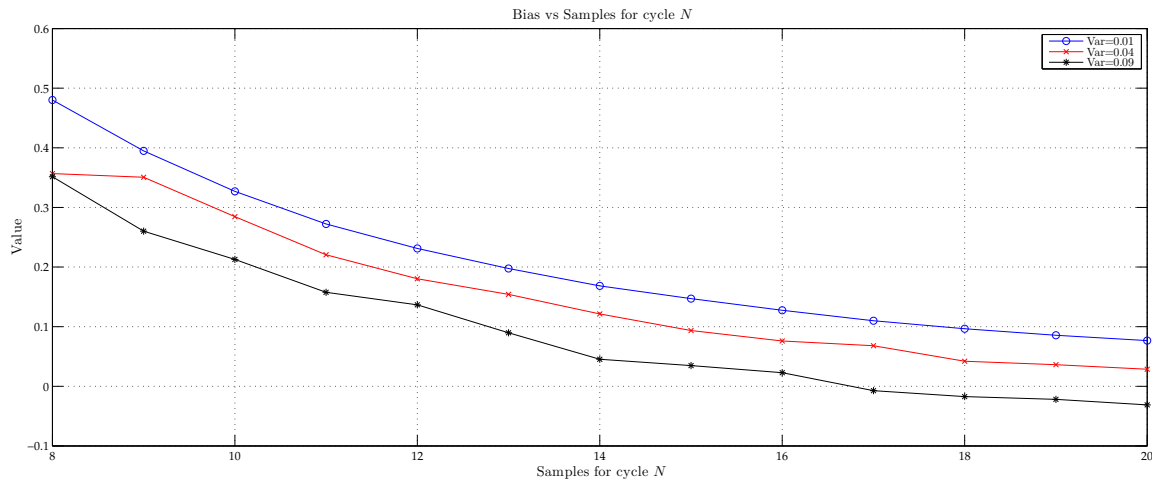


Figure 14. Bias versus the samples for cycle (N) for CLMS and CRLS algorithms for noise variance of 0.01, 0.04 and 0.09.

For all adaptive algorithms the mean-square error has reached the minimum value which means the cascade form does not increase this magnitude; even though, the number of stages could be enlarged.

References

- [1] B. Widrow and S. Stearns, *Adaptive signal processing*, 1st ed. Englewood Cliffs, NJ: Prentice-Hall, Inc, 1985.
- [2] L. Sibul, *Adaptive signal processing*, 1st ed. New York: IEEE press, 1987.
- [3] A. Sayed, *Adaptive filters*, 1st ed. New Jersey: Wiley-IEEE Press, 2008.
- [4] L. B. Jackson and S. L. Wood, "Linear prediction in cascade form," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 6, pp. 518–528, 1978.
- [5] P. Prandoni and M. Vetterli, "An FIR cascade structure for adaptive linear prediction," *IEEE Transactions on Signal Processing*, vol. 46, no. 9, pp. 2566–2571, 1998.
- [6] W. Orozco, "A cascade predictor filter of one-tap stages for adaptive linear prediction," Dec. 2005, Final project, New York Institute of Technology, Old Westbury, The United States.
- [7] J. Martinez and K. Nakano, "Cascade lattice IIR adaptive filter structure using simultaneous per-

- turbation method for self-adjusting SHARF algorithm,” in *SICE Annual Conference, 2008*. IEEE, 2008, pp. 2156–2161.
- [8] D. Shi and Y. Yu, “Design of discrete-valued linear phase FIR filters in cascade form,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 7, pp. 1627–1636, 2011.
- [9] R. Patnaik, V. Vandrasi, C. Madsen, A. Eftekhari, and A. Adibi, “Comparison of cascade, lattice, and parallel filter architectures,” *Journal of Lightwave Technology*, vol. 28, no. 23, pp. 3463–3469, 2010.
- [10] H. N. Apolo Castillo and A. E. Córdova Medina, “Modelación matemática y simulación de un filtro digital híbrido FIR adaptativo lineal óptimo,” 2010, Proyecto de graduación, Universidad Politécnica Salesiana, Cuenca, Ecuador.
- [11] A. Sayed, *Fundamentals of adaptive filtering*, 1st ed. New Jersey: Wiley-IEEE Press, 2003.